

REMARKS

Favorable reconsideration, reexamination, and allowance of the present patent application are respectfully requested in view of the foregoing amendments and the following remarks. Claims 1-18 and 21-26 are pending in the application.

Allowable Subject Matter

Applicant notes with appreciation the indication on page 2 of the Office Action that claims 19 and 20 are allowed.

35 U.S.C. § 103 Rejections

Claims 1-2, 4-9, 11-18 and 21-26 were rejected under 35 U.S.C. § 103 (a) as allegedly being unpatentable over Katseff et al. (U.S. Patent No. 5,822,537) in view of Dixit (U.S. Patent No. 5,260,783) and Joseph et al. (U.S. Patent No. 5,819,034). Claims 3 and 10 were rejected under 35 U.S.C. § 103 (a) as allegedly being unpatentable over Katseff in view of Dixit and Joseph et al. and in further view of Takahashi (U.S. Patent No. 5,739,865).

Regarding claim 1, the Examiner has correctly noted that the combination of Katseff et al. (hereafter "Katseff") in view of Dixit does not at least disclose a data extractor and wherein a number of frames within the extracted frame data corresponds to a load condition processed by said load processing device. However, the Examiner proposes to remedy these noted deficiencies by adding the newly cited Joseph et al.

patent (hereafter "Joseph"). Applicant respectfully submits that the Joseph reference fails to cure the above noted deficiencies. For example, the Examiner alleges that Joseph disclose a data extractor for extracting frame data from video data comprising frame data, wherein a number of frames within the extracted frame data corresponds to a load condition processed by said load processing device, as claimed by Applicant. Joseph does clearly not support this assertion by the Examiner. In contrast to the Examiner's assertions, Joseph discloses a data extractor that extracts data packets containing executable code from a data stream, as describe in column 3 lines 1-11 relied upon by the Examiner and reproduced below with emphasis added.

The inventors propose a distributed computing system in which a server computer continuously produces a data stream. This data stream acts a mass storage device for the client computers receiving it. This **data stream repetitively includes data representing a distributed computing application** in which the client computer may participate, including **executable code and data**. A transport mechanism, including a high speed, one-way, communication path, carries the data stream from the server to the client. The client receives the data stream, **extracts the distributed computing representative data and executes the distributed computing application**.

It is clear from the above-cited passage that the data extracted from the data stream is executable code and data for executing a distributed computing application. There is no teaching or suggestion in Joseph for a data extractor for extracting frame data from video data, as claimed by the Applicant.

Further, the data extractor in Joseph is in client side. Therefore it does not contribute to the data distribution at server side, i.e., the video data distribution device. Still, further, there is no teaching or suggestion in Joseph that discloses wherein a

number of frames within the extracted frame data corresponds to a load condition processed by said load processing device. In direct opposition to this characterization, Joseph requires that the data eventually extracted by the data extractor constitute an entire executable code (i.e., all related packets must be extracted). Joseph clearly teaches to use the video signal as a transport mechanism on which to piggyback application programs to the client computers. This technique requires that the applications are multiplexed onto the transport stream in packets. Additionally, Joseph discloses that these packets are repetitively inserted into the transport stream to form the "mass storage" capability that Joseph teaches. Essentially, Joseph discloses multiplexing a repetitive series of packets of executable programs and related data on the transport stream and selectively downloading (extracting) the desired program to the client computer. This aspect is clearly defined in column 4, line 60 to column 5, line 9 of Joseph, which is reproduced below with emphasis added.

In operation, the server 10 produces a continuous data stream in the form of a stream of packets for the client computers 20. The server 10 **repetitively inserts a packet, or successive packets, containing data representing the distributed computing application**, including at least one executable code module, into the data stream. This code module contains executable code for the client computers 20. The data receiver in, for example, client computer 22, continuously monitors the packets in the data stream on transport mechanism 30. When a packet including identification information indicating that it contains the code module (or a portion of the code module) required by the client computer 22 is present in the data stream, the client computer 22 detects its presence, **extracts the code module (or the portion of the code module) from that packet and stores it in the main memory. When the code module is completely received, the client computer 22 begins to execute it.**

Contrary to the Examiner's allegations, Joseph does not disclose a data extractor for extracting frame data from video data comprising frame data, wherein a number of frames within the extracted frame data corresponds to a load condition processed by said load processing device. Instead, Joseph discloses a means for distributing applications to client computers multiplexed into a data stream.

As stated in MPEP § 2143.01, to establish *prima facie* obviousness of a claimed invention, all the claim limitations must be taught or suggested by the prior art. *In re Royka*, 490 F.2d 981, 180 USPQ 580 (CCPA 1974). "All words in a claim must be considered in judging the patentability of that claim against the prior art." *In re Wilson*, 424 F.2d 1382, 165 USPQ 494, 496 (CCPA 1970). The combination of Katseff, Dixit and Joseph fails to disclose the features of Applicant's claimed combinations at least for the reasons noted above. Therefore, these references do not render Applicant's claimed combinations obvious as alleged by the Examiner. Accordingly, Applicant respectfully requests reconsideration and withdrawal of this rejection.

Regarding remaining independent claims (i.e., 4, 6-9, 11, 16, 23, and 26) and the remaining dependent, the Examiner has attempted to use similarly misinterpreted the teachings of the Joseph et al. as outlined above in regard to the features of claim 1. Accordingly, Applicant respectfully requests the Examiner to reconsider and withdraw these rejections.

Further, as discussed extensively in Applicant's prior responses dated September 19, 2002, the combination of Katseff in view of Dixit fails to teach Applicant's claimed combinations. Therefore for brevity, Applicant expressly incorporates the prior

arguments presented in the September 19, 2002, response without a literal rendition of those arguments in this response.

CONCLUSION

For at least the foregoing reasons and the reasons set forth in Applicant's response of September 19, 2002, it is respectfully submitted that all claims are distinguishable over the applied art.

Should there be any outstanding matters that need to be resolved in the present application, the Examiner is respectfully requested to contact Mark E. Olds, Reg. No. 46,570, at the telephone number of the undersigned below, to conduct an interview in an effort to expedite prosecution in connection with the present application.

If necessary, the Commissioner is hereby authorized in this, concurrent, and future replies, to charge payment or credit any overpayment to Deposit Account No. 02-2448 for any additional fees required under 37 C.F.R. §§ 1.16 or 1.17; particularly, extension of time fees.

Respectfully submitted,

BIRCH, STEWART, KOLASCH &, BIRCH, LLP

By: 

Michael K. Mutter, Reg. No. 29,680
P.O. Box 747
(703) 205-8000
Falls Church, VA 22040-0747

MKM/MEO/ylw/lab
0925-0142P